

ASSIGNMENT-1:DATABASE DESIGN

Database Domain:

Real estate property listings database

Description of Database:

The database is designed to support the process of buying, selling, and renting real estate properties. It maintains records of available properties, property owners, potential buyers and renters, and the agencies managing these properties.

Each property has various details, including its ID, Type (Apartment, House, Commercial) , status(Available, Sold, Rented), size, price and address, which consists of house number, street, city and PIN.

Each property is owned by an owner, and each owner must own at least one property. The owner's details include ID, name, email IDs, contact number and address, which consists of house number, street, city and PIN.

A Buyer/Renter is an individual interested in purchasing or renting a property. The Buyer/Renter's details include ID, name, email IDs, contact number and address, which consists of house number, street, city and PIN. Each buyer or renter can be interested in multiple properties, and each property can be of interest to multiple buyers or renters. It is necessary for a buyer to be interested in at least one property.

Real estate agencies play a key role in managing properties, verifying properties and assisting clients. An agency's details include name, email IDs, contact number, ID, address (house number, street, city, and PIN) and license number. Each agency verifies multiple properties, and each property can be verified by a multiple agency.

Each buyer or renter can subscribe to multiple agencies, and each agency can have multiple buyers or renters subscribed to receive notifications about new properties. It is must that a Buyer/Renter be subscribed to at least one agency. Each agency employs multiple agents who manage multiple properties, and multiple agents can manage a single property. It is given that an agent belongs to exactly one agency. Agent details include name, email IDs, contact number, ID and commission rate. Each agent can handle multiple buyers or renters, and each buyer or renter must be assisted by at least one agent.

An agent facilitates the transaction of a property by helping the Buyer or Renter navigate the process. Each transaction includes details such as transaction ID, payment mode (Cash, UPI), property value, date and time. Each

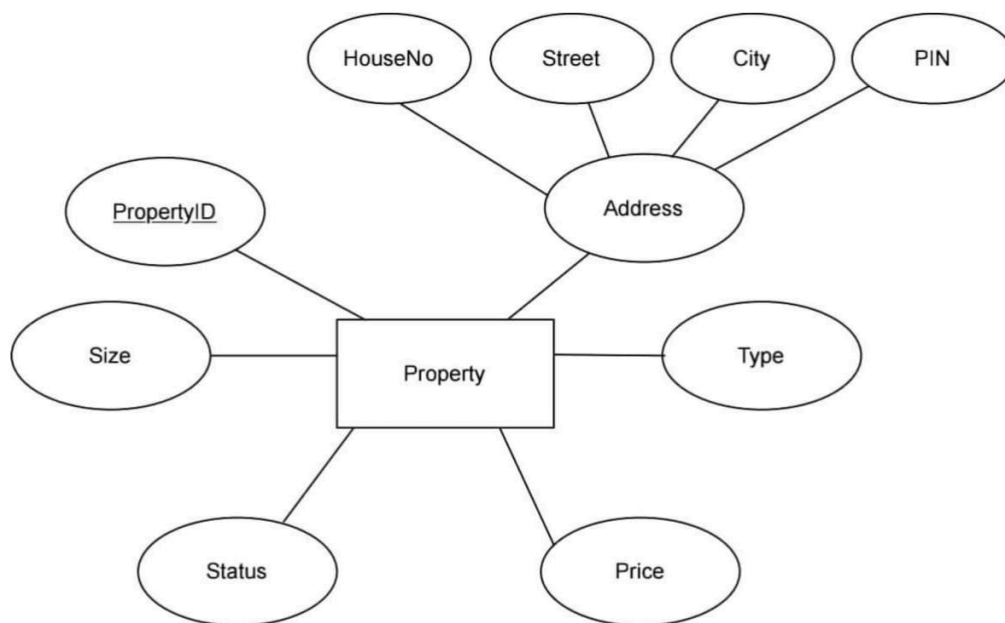
agent can have multiple owners as clients, and each owner can be a client of multiple agents.

Each buyer or renter can schedule a visit to a property with the help of an agent. The schedule contains details such as visit ID, time, date and status (Scheduled, Completed, Cancelled).

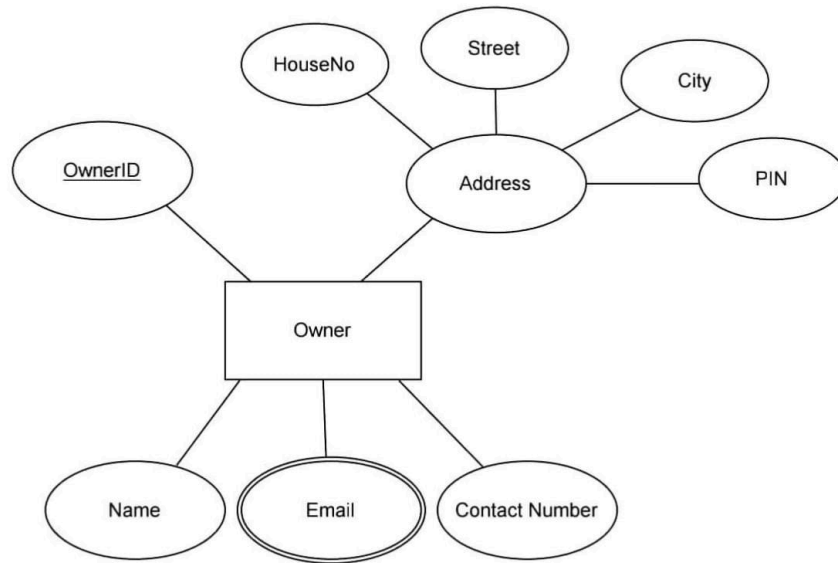
Overall, this system simplifies and organizes real estate transactions, making it easier for property owners, buyers/renters, and real estate professionals to connect and interact efficiently.

Entity-Relationship Model

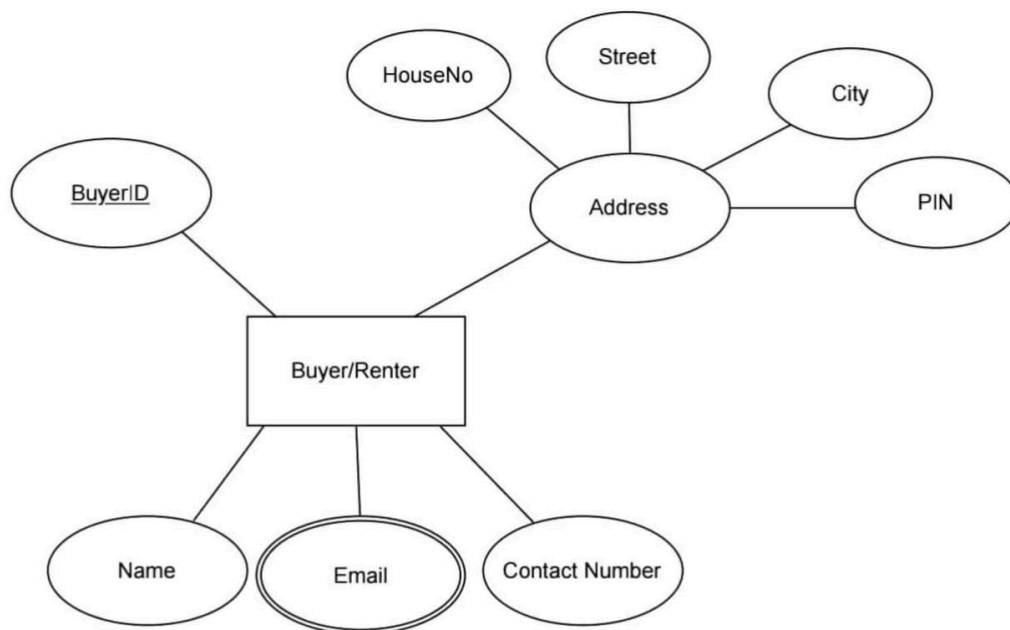
Property



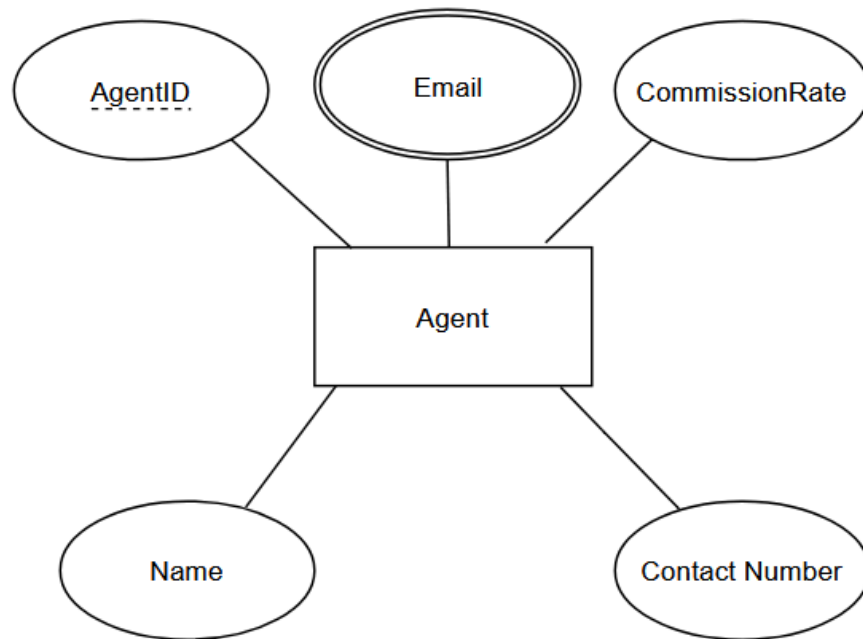
Owner



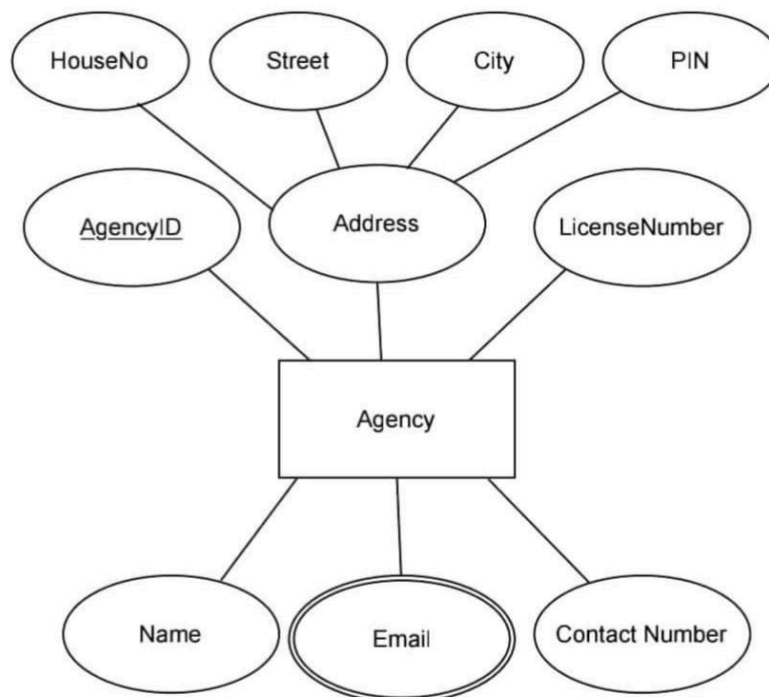
Buyer/Renter



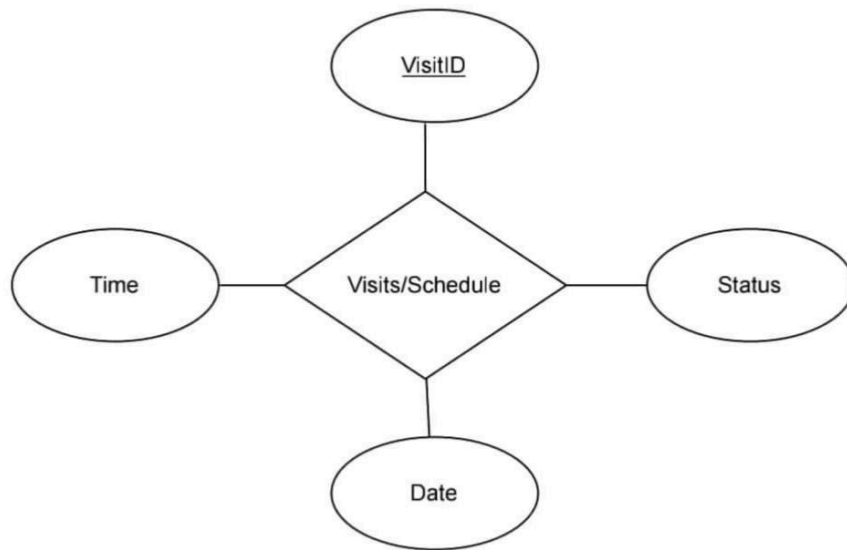
Agent



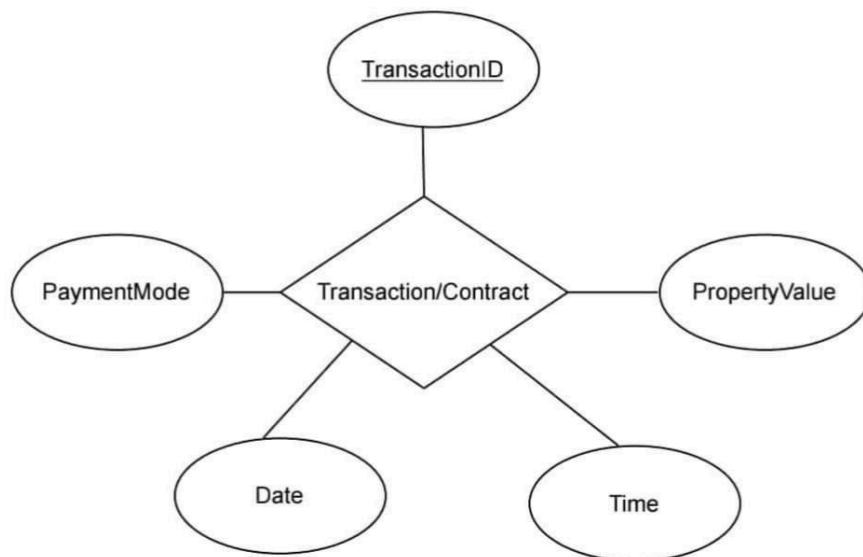
Agency



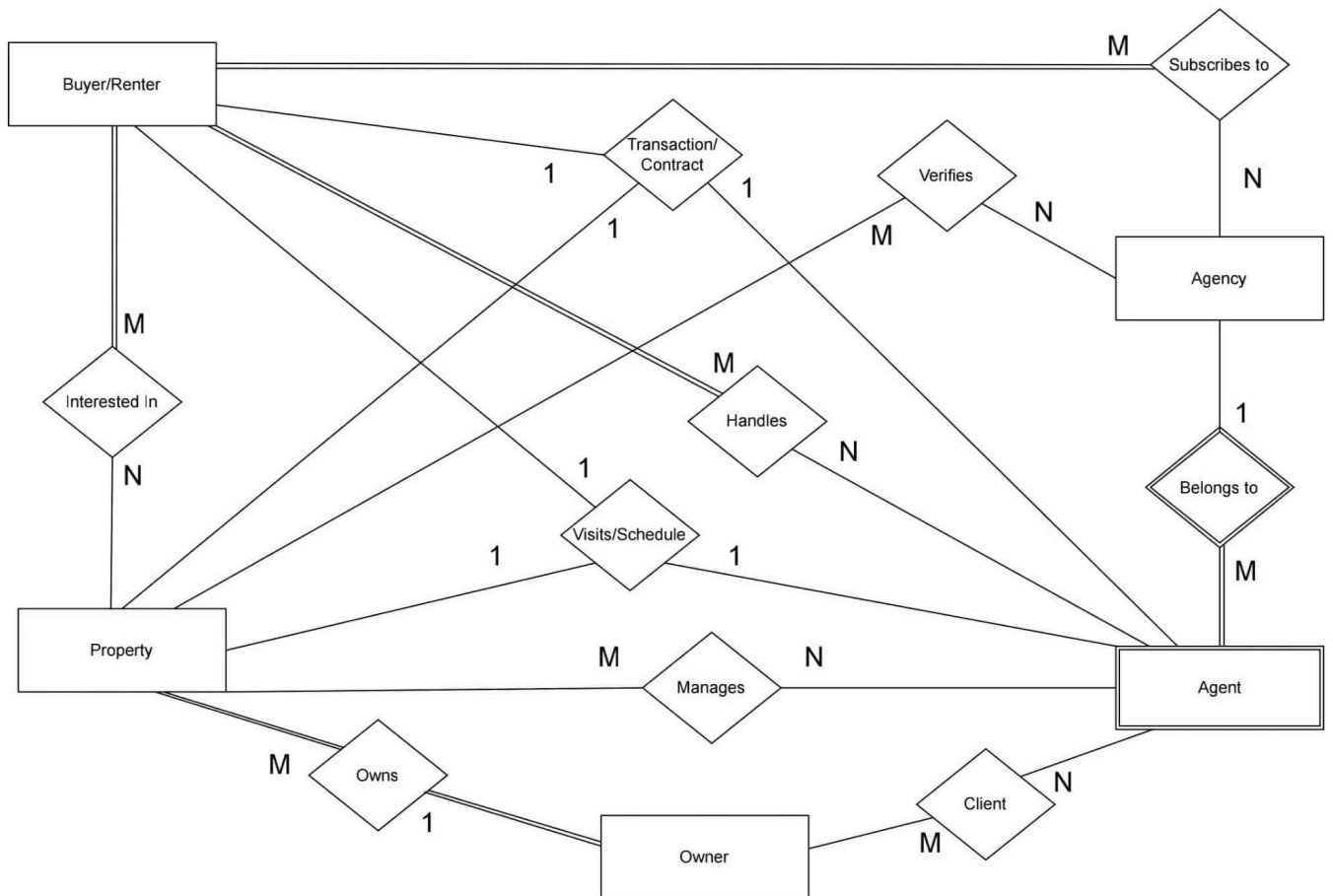
Visit/Schedule



Transaction/Contract



E/R Diagram showing relationships



Conversion of ER diagram to Relational Database

Step 1:

There are 5 entities in this ER diagram namely Property, Owner, Buyer, Agent and Agency.

The first process was to add simple attributes and components of composite attributes (which was the “address” attribute in this case) for these entities to their respective relational tables.

We have represented them in the center of the diagram.

Important Note. Handling the weak entity and relation.

Agent here is a weak entity. So for the primary key of Agent we include both partial key AgentID and key of owner entity which is AgencyID.

Step 2:

Dealing with multi value attribute i.e email id for buyer, owner and agent. Since the database is atomic we have dealt with this by adding a new table with the key as mail id AND primary key of the entity consisting the multi value attribute.

They have been represented to the left of the diagram.

Step 3:

Considered the relations of Visits/Schedules as separate tables (like the third method used for M:N relations) rather than adding them to one of the entities because

1. It was a three-way relation
2. The relation itself has attributes and a key. Felt it would be a better design choice to add them as separate tables. Would save space in the entity set.

Added its simple attributes and key. Apart from it we need to add the primary key of the entities it is in relation with namely Agent, Property and Buyer for both transactions and visits relations.

Since the TransactionID and VisitID themselves are a key we do not need to include the primary keys from Agent, Property and Buyer in the key.

Step 4:

Dealing with 1:1 relations and 1:N relations.

1:1 – There was no particular 1:1 relation apart from the above two.

1:N – You add the primary key of the entity on the 1 side of the relationship to the N side entity's table.

All the primary keys added as such are represented at the end for better understanding.

The relationships which needed this processing were:

1. Property -> Owner. (property owned by only one owner, taken for convenience. One owner can own many properties)
Hence added OwnerID to Property relation
2. Agency -> Agent (an agent is part of only one agency while agency can have multiple agents)
Hence added AgencyID to Agent relation

They have been represented to the left of the diagram (along with multivalued attribute tables).

Step 5:

Dealing with M:N relations. There are 6 M:N relations namely:

1. Verifies : Property and Agency
PropertyAgency relation
2. Subscribes to: Buyer and Agency
BuyerAgency relation
3. Handles: Agents and Buyers
BuyerAgent relation (keep in mind agent being a weak entity we have to take the primary key and not the partial key)
4. Interested In: Buyers and Property
PotentialBuyers relation
5. Manages: Agents and Property
PropertyAgent relation to the right.
6. Finally the Client : Owner and Agent
OwnerAgent relation

For such we used the method described in class i.e we have added the primary key of both the entities in our new relation table and this combination of both will be the key.

The obtained relations are represented to the right of the diagram.

Please find attached link for relation schema diagram.

[Relation Schema Diagram](#)